

1/10

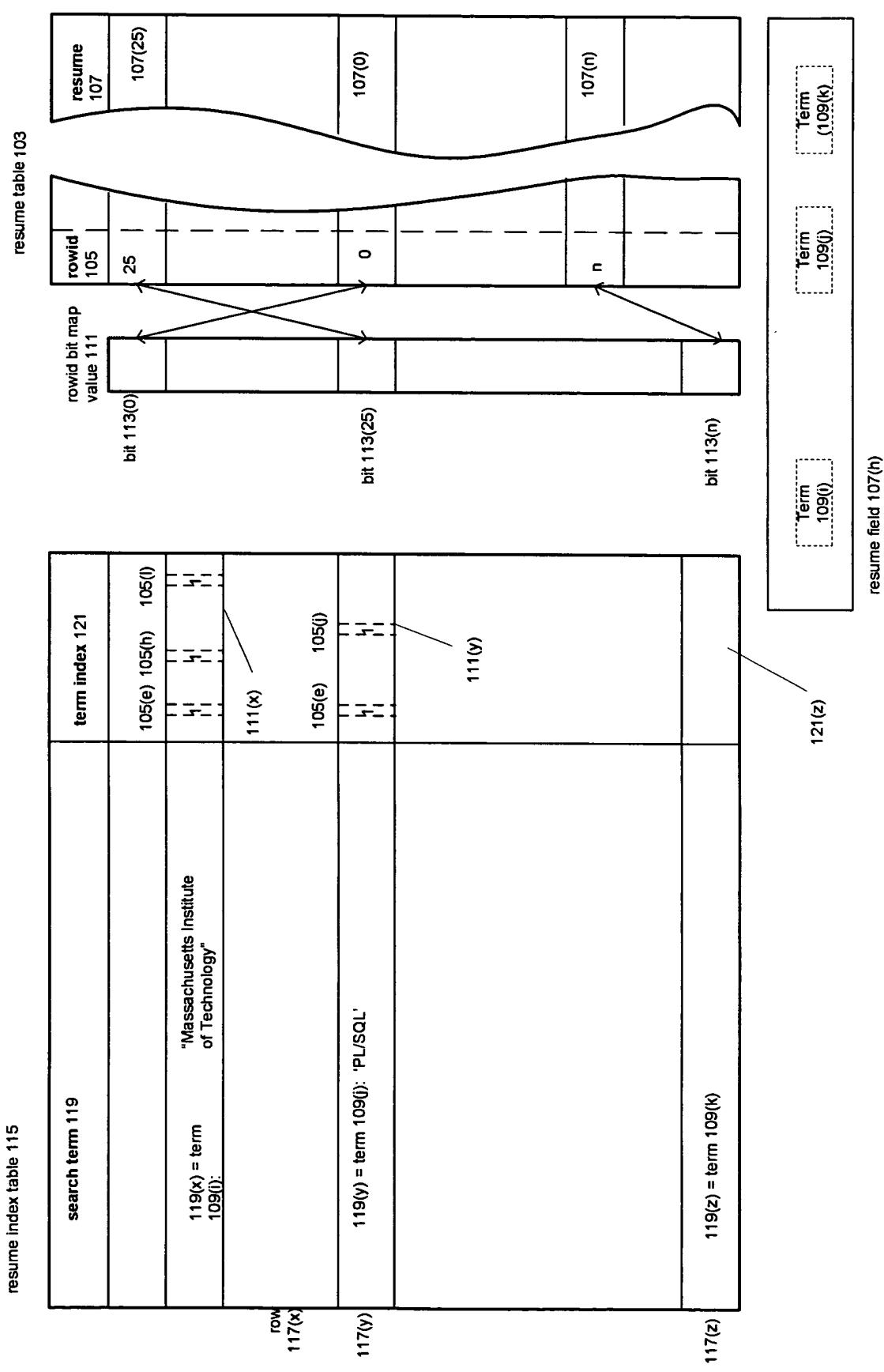


Fig. 1

2/10

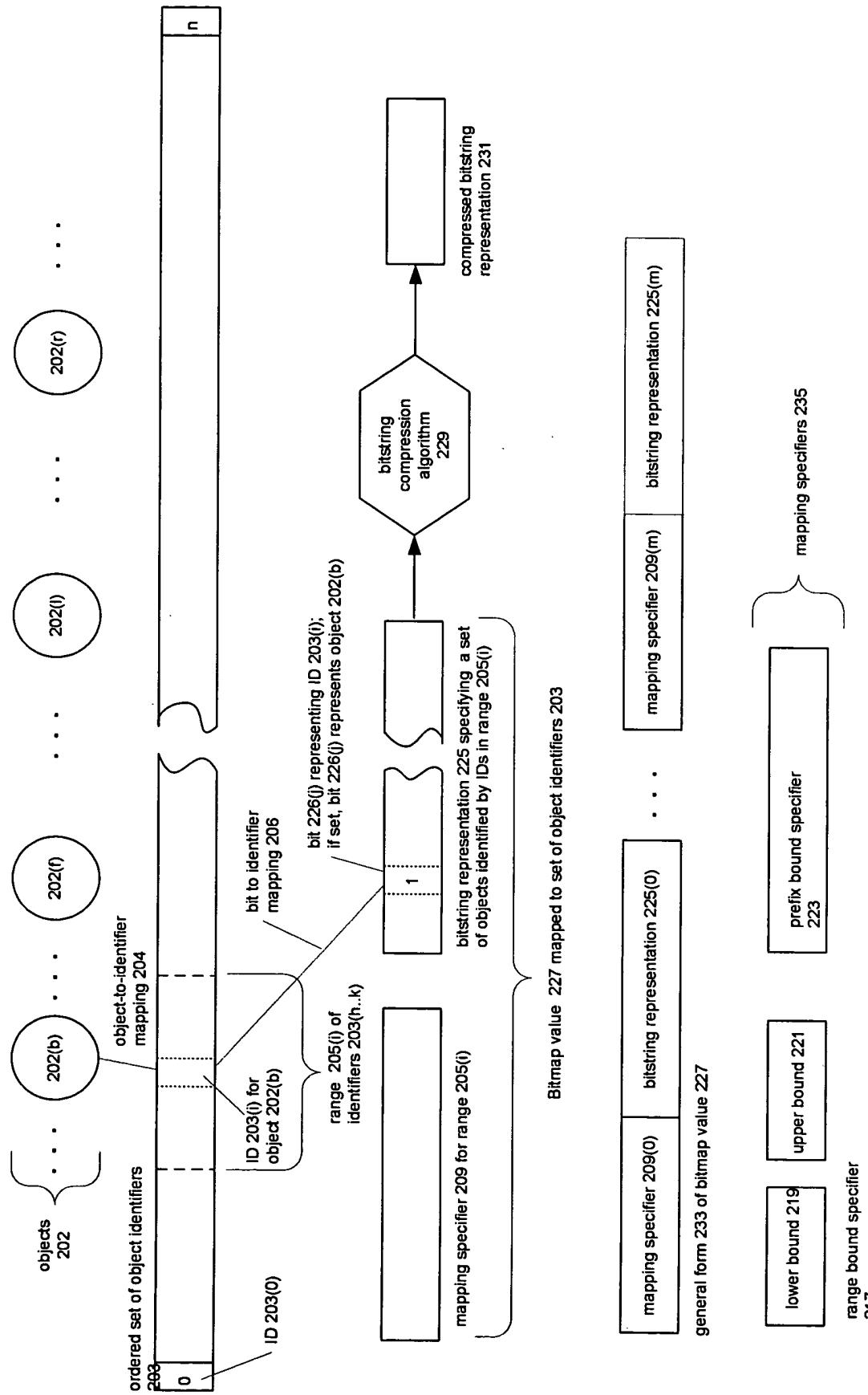
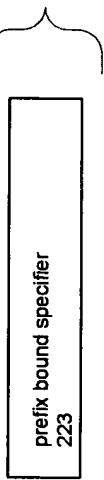


Fig. 2



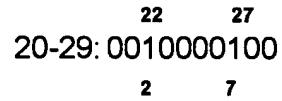
mapping specifier 209(0)	bitstring representation 225(0)	...	mapping specifier 209(m)	bitstring representation 225(m)
--------------------------	---------------------------------	-----	--------------------------	---------------------------------

mapping specifier 209(0)	bitstring representation 225(0)	...	mapping specifier 209(m)	bitstring representation 225(m)
--------------------------	---------------------------------	-----	--------------------------	---------------------------------

**Bitmap values mapped onto identifier range 20-29:**

**Mapping specifier: 20-29**      **Bitstring:**      

303

**Bitmap value bitmapval\_1**      

305

**bitmap-to-set operation**

bitmap\_to\_set(bitmapval\_1)

**result**

{object\_id(22), object\_id(27)}

307

**set-to-bitmap operation**

bitmapval\_2 := set\_to\_bitmap({object\_id(20), object\_id(22),  
                                  object\_id(24)})

309

**result**

bitmapval\_2 = 20-24:10101

**bitmap-to-count operation**

intval := bitmap\_to\_count(bitmapval\_1)

**result**

intval = 2

intval := bitmap\_to\_count(bitmapval\_2)

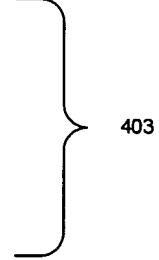
**result**

intval = 3

311

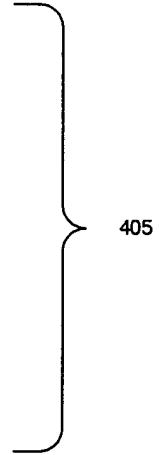
4/10

**Bitmap values:** bitmapval\_1      20-29:0010000100  
                                      22      27  
                                      2          7  
  
                                      20 22 24  
bitmapval\_2      20-24:10101  
                                      0 2 4



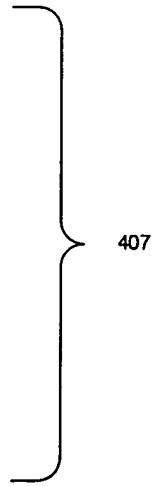
**object exists in bitmap operation**

```
intval := object_exists(bitmapval_1,object_id(26))  
  
result  
  
intval = 0  
  
intval := object_exists(bitmapval_1, object_id(22))  
  
result  
  
intval = 1
```



**bitmap equals**

```
intval := equals(bitmapval_1, set_to_bitmap(  
                                      {object_id(22),object_id(27)}))  
  
result:  
  
intval=1  
  
intval := equals(bitmapval_1, bitmapval_2)  
  
result:  
  
intval=0
```



401

Fig. 4

**Bitmap values:**

bitmapval_1	20-29: 0010000100	503
	22      27 2        7 20 22 24 0 2 4	
bitmapval_2	20-24: 10101	

**bitmap-AND operation**

```
result_bitmap_val := bitmap_and(bitmapval_1, bitmapval_2)
```

**result**

```
result_bitmap_val = 20-29:0010000000
```

**bitmap-OR operation**

```
result_bitmap_val := bitmap_or(bitmapval_1, bitmapval_2)
```

**result**

```
result_bitmap_val = 20-29:1010100100
```

**bitmap-XOR operation**

```
result_bitmap_val := bitmap_xor(bitmapval_1, bitmapval_2)
```

**result**

```
result_bitmap_val = 20-29:1000100100
```

**bitmap-minus operation**

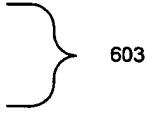
```
result_bitmap_val := bitmap_minus(bitmapval_1, bitmapval_2)
```

**result**

```
result_bitmap_val = 20-29:0000000100
```

**Bitmap value:** bitmapval\_1

22	27	
0010000100		
2	7	



### bitmap-insert operation

```
bitmap_insert(bitmapval_1, {object_id(28), object_id(33)})
```

#### result

22	2728	33	
bitmapval_1 = 20-33:00100001100001			
2	7 8	13	

605

### bitmap-delete operation

```
bitmap_delete(bitmapval_1, {object_id(27), object_id(28)})
```

#### result

22	27	
bitmapval_1 = 20-29:0010000000		
2	7	

607

601Fig. 6

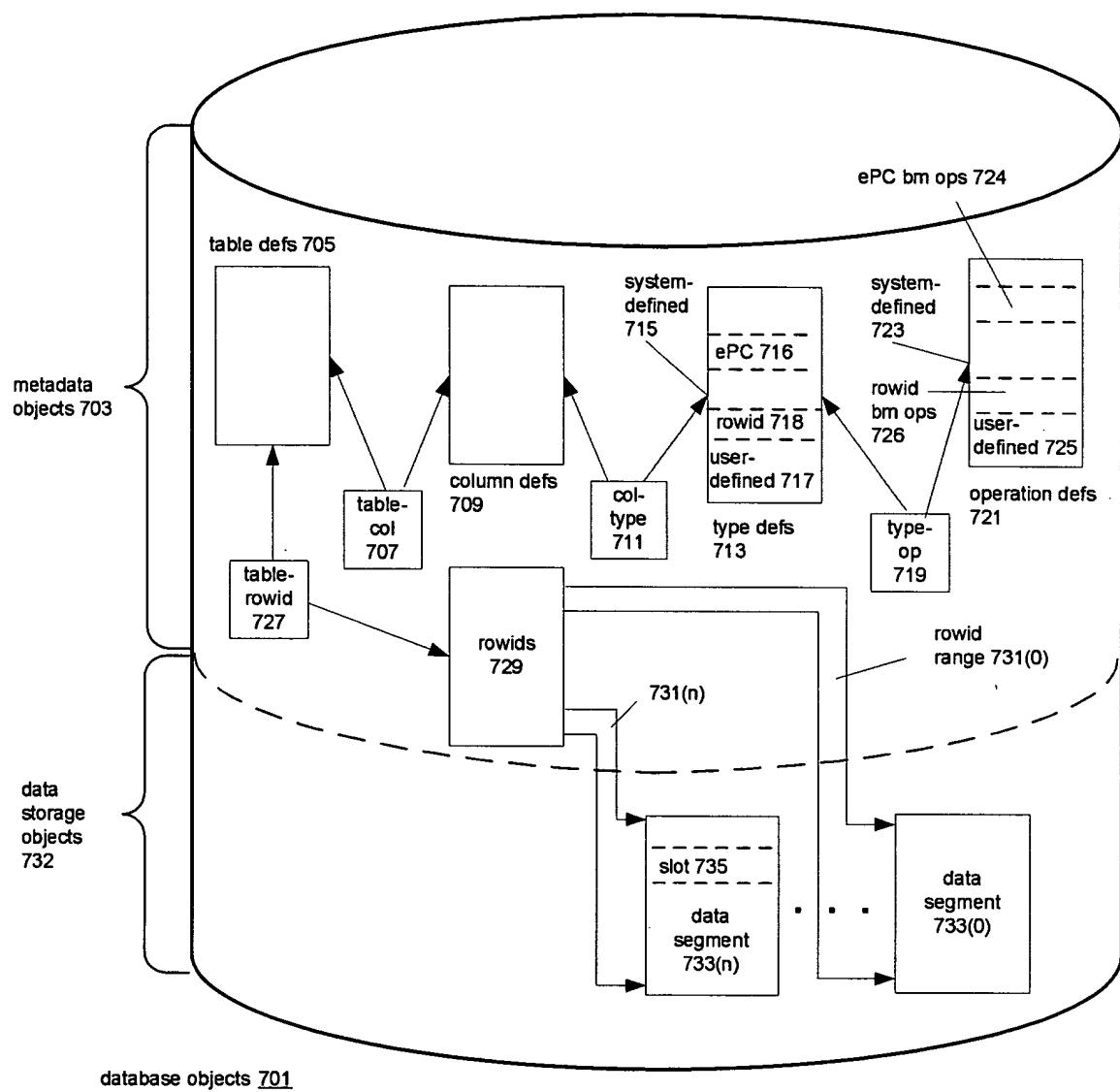


Fig. 7

8/10

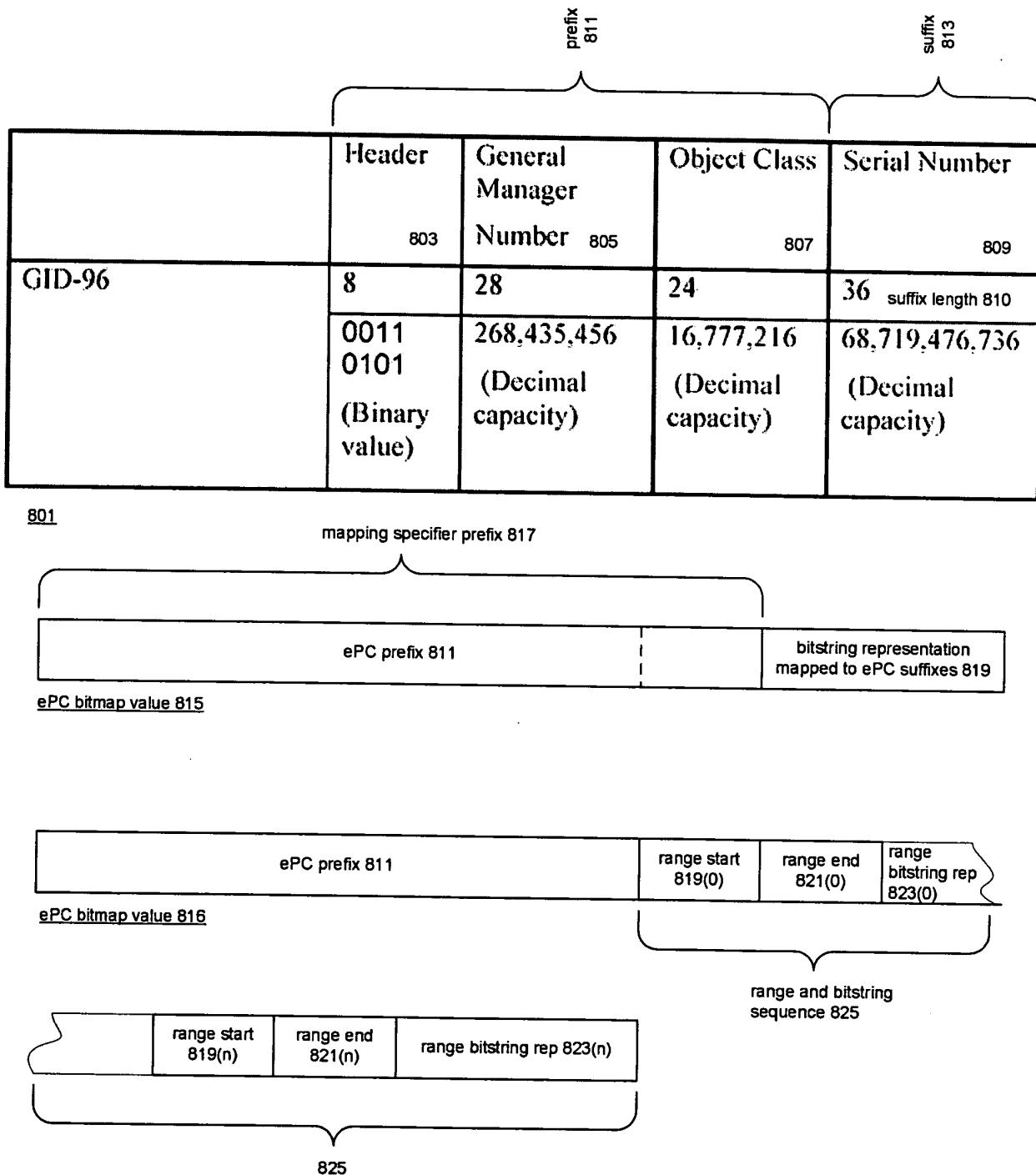


Fig. 8

9/10

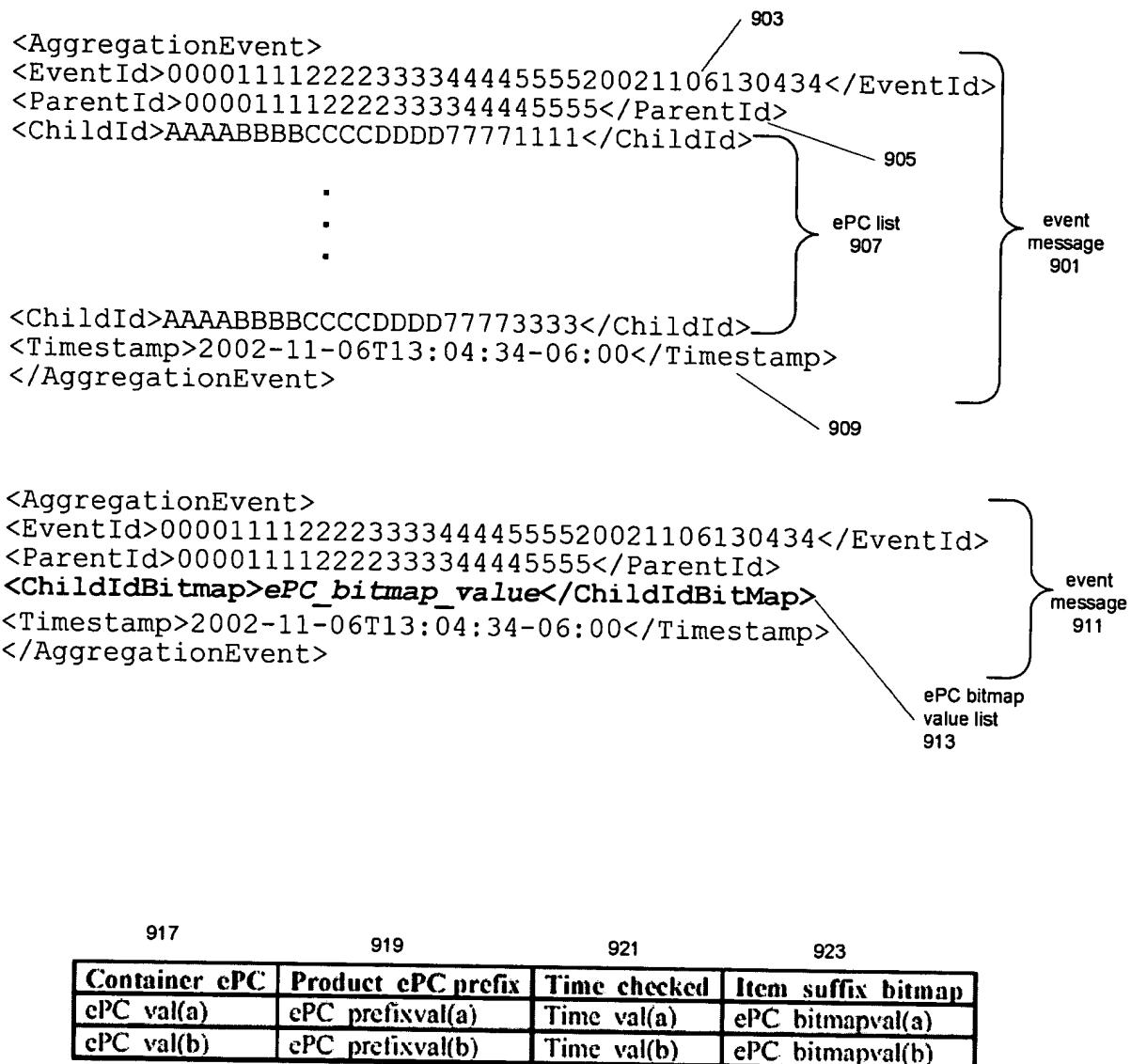


Fig. 9

10/10

data object # 1005	datafile # 1007	data block # 1009	row # 1011
--------------------	--------------------	-------------------	---------------

rowid 1003

rowid range start 1023(a)	rowid range end 1025(a)	range bitstring representation 1027(a)
------------------------------	----------------------------	---



mapping specifier 1021(a)

1013(n)	1025(n)	1027(n)
---------	---------	---------

rowid bitmap value 1019

1001

Fig. 10